

**软件杯作品说明文档**



**学生姓名　 陆铮 赵倩锐 王沛然**

**专　　业 计算机科学/网络空间安全**

**指导教师 黄诚**

**二〇二一 年 6 月 25 日**

目录

[1. 作品背景 3](#_Toc1995)

[2. 作品技术 4](#_Toc18007)

[3. 作品框架 4](#_Toc7429)

[4. 作品测试 4](#_Toc25462)

[4.1. 测试环境 4](#_Toc27747)

[4.2. 测试步骤 4](#_Toc18897)

[4.3. 测试结果 4](#_Toc25111)

[4.4. 测试结果分析 4](#_Toc31743)

[5. 作品使用说明 4](#_Toc26828)

## 作品背景

随着互联网技术的快速发展，在各类社交媒体中作为文本、图片、音频和视频等各类信息都在不断产出，其中最容易、最方便获取的当属文本形式的信息。像新闻标题、微博博文和在线评论这种中文短文本信息，最主要特点就是稀疏性，而且其中所包含的有效信息是非常少的，样本的稀疏性直接导致了在文本分类时很难找到相对准确和有用的信息。另外，短文本信息在互联网更新速度时非常快的，但是这些文本信息在信息检索性、个性化新闻推荐、关系抽取和用户意图分析等各个领域的研究中都是非常重要的研究数据。因此如何快速有效地获取这些文本信息中有价值的特征数据变得越来越重要。

随着社交网络和移动终端技术的飞速发展，文本信息作为互联网中比较直接的媒体形式，所占的比重越来越大，这些在线的文本数据主要特点是高实时性和高复杂性。互联网的发展，使得每时每刻都有无数用户在使用这些信息数据，而如何使用文本处理技术提取这些信息数据中有价值的特征，然后分析每个用户的兴趣爱好和关注点，已经成为目前互联网时代的主要研究方向。

文本分类作为文本信息处理中最关键的技术之一，目前主要应用于信息检索、信息监管和知识挖掘等领域上。目前应用在文本分类主要有:K临邻(K-NearestNeighbor）算法、朴素贝叶斯、支持向量机(SVM）和用 BP (Back Propagation)神经网络等分类模型，并且均取得了良好的效果。当然为了能够让文本分类的效果比原来更好，有些学者对这些传统的文本分类算法做了进一步改进。但相对文本分类的任务来说，这些分类方法都无法解决以下的问题:分类算法容易出现局部最优；分类模型中数据的维数太大造成模型计算复杂度过高，从而导致模型崩溃；模型无法获取数据的关键特征导致出现过拟合现象；模型无法学习到足够的特征从而导致其在多分类任务上无法具有较好的泛化性。

深度学习的发展，对自然语言处理的研究具有非常大意义的影响。深度神经网络优势在于使用了多层的非线性映射结构，从而有效的克服了浅层网络中的各种缺陷。深度神经网络使用多层结构训练时，不单单能够学习到更多的特征，同时不会使用大量的参数造成过多的复杂的计算。另外深度学习在处理分布式特征的数据时，选择将低层的特征进行组合，映射成为高层的特征，也就是利用逐层学习的方法获得输入数据更多的特征例。

## 作品技术

### 模型ERNIE 框架

补一下自己部分的框架介绍

### 后端Flask框架

#### Web Framework(WF)

Web Framework是一种支持开发Web应用，包括Web服务，Web资源和Web API的集成框架。它提供了一种在万维网上搭建Web应用服务的标准套路。现在，一般的Web开发框架都提供了数据库和会话的管理功能，也集成了一些常用模板。在生产开发中，无论是动态网站还是静态页面，WF都展示了良好的适应性。

#### Flask

Flask是一款基于Python语言的微型网页开发框架。这款开源框架最大的特点是短小精悍，它的源码只有万行左右，没有默认使用的数据库层和窗体验证工具等等，但是Flask的可扩展性很好地解决了这一个问题。开发者可以通过Flask-extension可扩展插件来增添新功能，包括数据库访问，文件上传，身份验证等等。



Flask所扮演的角色

#### Why Flask?

我们为什么打算选用Flask作为数据可视化的前端开发框架，而不选用其他的Web开发框架或者直接使用传统的HTML,CSS,Javascript三件套呢？主要原因有以下几点:

1. 效率提升

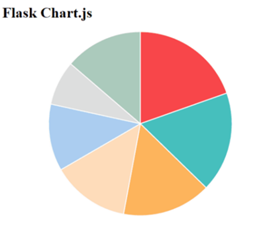
搭建Web应用往往需要进行大量无聊繁复的操作，诸如构建终端逻辑，编写面向用户的图形化用户界面，此外还需要放置于因特网中使用户能够访问你的服务。Web框架旨在实现普遍的Web服务功能，可以减少冗余操作，提升开发效率。

1. 安全性佳

对于前端而言，有许多渗透方式，如XSS（跨站脚本攻击），CSRF（跨站请求伪造）等等。这些入侵手段轻则伪造用户身份，窃取用户信息，严重则可能导致被提取管理员权限。客观而言，我们也是第一次进行Web应用开发，经验严重不足，尤其是安全问题很可能在编写代码的时候忽视。幸运的是，Flask集成框架为开发者提供了一定的安全保护。比如它自动转义了所有值，避免了模板导致的XSS攻击。

1. 短小精悍

可以说，“麻雀虽小，五脏俱全”是对Flask的一个中肯评价。Flask最大的特点是微小，其代码更加简洁而且弹性好，这也为程序员提供了更多的自由度和扩展空间。相比自行重新编写传统的HTML和CSS的冗长代码，使用Flask的模板往往效率更高，更加整洁美观。另外Flask是近年来较新的Web框架，更新频繁，与时俱进。因此综上所述，Flask是一款极其适合初学者快速上手的开发框架。



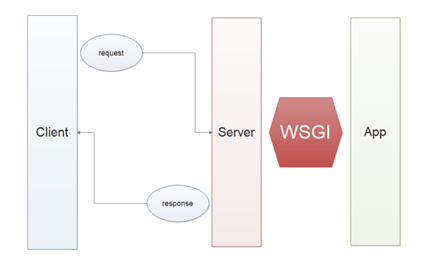
利用Flask模板绘制的饼图

#### 与其他框架的比较

Django也是一个基于Python的Web框架。与Flask相比，Django代码更加复杂，而且开发时间太早，因此Flask轻量级，灵活性更好，适宜新手快速上手。

在开源代码平台Github上，Flask所获star数量也高于Django，开发者的整体偏爱程度可见一斑。

#### 框架原理



Flask架构

1. Web应用原理

Web应用的本质分为4步：1.浏览器向服务器发送请求。2.服务器接受客户端请求，并解析。3.服务器端把HTML作为响应体发送给浏览器。4.浏览器获取响应体渲染网页。

1. WSGI

WSGI，全称Web Server Gateway Interface，是一种服务器端与Web应用程序之间通信的规范。在客户端和服务器端WSGI负责协议的转化，其将Web组件分为三部分:Web服务器、Web中间件、Web应用程序。当服务器接收到HTTP请求时，会按照WSGI协议解析成Request对象并调用WSGI Application，最后将响应返回给浏览器。

1. Werkzeug

Python下有一个库Werkzeug，是Python中的WSGI规范的实用函数库。实现Flask底层的WSGI规范的库就是Werkzeug。

WSGI规定了服务器和application之间的通信，其规定了一个application接口，参数包含一个二元组（env, func），其中env是环境设置的字典，包含了请求的所有信息，func是WSGI处理完毕后开始调用的函数。

### 前端XXX框架

补一下自己部分的框架介绍

## 作品框架（我来弄，正在画框架图）

## 作品测试（弄完写，14号必须弄完哦）

### 测试环境

### 测试步骤

### 测试结果

### 测试结果分析

## 作品使用说明